

File ..



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/620,714	07/20/2000	Luke Matthew Browning	AUS9-2000-0277-US1	3354
35525	7590	10/17/2003	EXAMINER	
DUKE W. YEE CARSTENS, YEE & CAHOON, L.L.P. P.O. BOX 802334 DALLAS, TX 75380			STEELMAN, MARY J	
			ART UNIT	PAPER NUMBER
			2122	9
DATE MAILED: 10/17/2003				

Please find below and/or attached an Office communication concerning this application or proceeding.

Advisory Action

Application No.

09/620,714

Applicant(s)

BROWNING ET AL.

Examiner

Mary J. Steelman

Art Unit

2122

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

THE REPLY FILED 03 September 2003 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE. Therefore, further action by the applicant is required to avoid abandonment of this application. A proper reply to a final rejection under 37 CFR 1.113 may only be either: (1) a timely filed amendment which places the application in condition for allowance; (2) a timely filed Notice of Appeal (with appeal fee); or (3) a timely filed Request for Continued Examination (RCE) in compliance with 37 CFR 1.114.

PERIOD FOR REPLY [check either a) or b)]

- a) ☒ The period for reply expires 3 months from the mailing date of the final rejection.
- b) ☐ The period for reply expires on: (1) the mailing date of this Advisory Action, or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection. ONLY CHECK THIS BOX WHEN THE FIRST REPLY WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

1. ☐ A Notice of Appeal was filed on _____. Appellant's Brief must be filed within the period set forth in 37 CFR 1.192(a), or any extension thereof (37 CFR 1.191(d)), to avoid dismissal of the appeal.
2. ☐ The proposed amendment(s) will not be entered because:
- (a) ☐ they raise new issues that would require further consideration and/or search (see NOTE below);
- (b) ☐ they raise the issue of new matter (see Note below);
- (c) ☐ they are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
- (d) ☐ they present additional claims without canceling a corresponding number of finally rejected claims.

NOTE: _____

3. ☐ Applicant's reply has overcome the following rejection(s): _____.
4. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
5. ☒ The a) ☐ affidavit, b) ☐ exhibit, or c) ☒ request for reconsideration has been considered but does NOT place the application in condition for allowance because: See attached text.
6. ☐ The affidavit or exhibit will NOT be considered because it is not directed SOLELY to issues which were newly raised by the Examiner in the final rejection.
7. ☐ For purposes of Appeal, the proposed amendment(s) a) ☐ will not be entered or b) ☐ will be entered and an explanation of how the new or amended claims would be rejected is provided below or appended.

The status of the claim(s) is (or will be) as follows:

Claim(s) allowed: _____.

Claim(s) objected to: _____.

Claim(s) rejected: _____.

Claim(s) withdrawn from consideration: _____.

8. ☐ The proposed drawing correction filed on _____ is a) ☐ approved or b) ☐ disapproved by the Examiner.
9. ☐ Note the attached Information Disclosure Statement(s) (PTO-1449) Paper No(s). _____.
10. ☐ Other: _____

Mary Steelman
Examiner

Wei Zhong
WEI ZHONG
Primary Patent Examiner

SUPPLEMENT TO ADVISORY ACTION, PAPER #9

In the remarks, the applicant has argued substantially that:

- 1) Regarding claim 1, the Kato reference does not teach the claimed step of retrieving the stored process state in response to a predefined event.

Examiner's response:

- 1) See col. 4, lines 20-23, "If a state restoration command is inputted, ..." which reads on the claim language, "in response to a predefined event". Col. 5, lines 35-48 does not imply that a user manually selects files, but rather states, "...when the debugged state is to be restored, by referring to the information, the state storage file can be determined readily from among existing state storage files.

In the remarks, the applicant has argued substantially that:

- 2) Regarding claim 6, the Kato reference fails to teach the claimed step of "reinitiating debugging from the stored process state, wherein the process has control over at least one child

Art Unit: 2122

process and the process state includes a process descriptor for each of the at least one child process.”

Examiner's response:

2) Kato does provide a multi-process environment (not in the claim language for claim 6, but included in the claim language for claims 28 & 29 only). Applicant fails to specify a parent process in relation to a child process, inferring any process (not specifically a parent process) having control over a child process. Claim 6 does not recite the limitation of “maintaining debug information for more than one running process”, as inferred on page 13, first paragraph of Amendment B.

In the remarks, the applicant has argued substantially that:

3) Regarding claim 26, the Kato reference does not teach the claimed steps of “retrieving the stored process state in response to a predefined event”...”wherein the predefined event is a checkpoint”.

Examiner's response:

3) Checkpointing is merely noting status or state of the computing environment at a point in time. Col. 6, lines 33-35, “...intended debugged state can be restored can be retrieved...and

Art Unit: 2122

restoration processing of a debugged state is facilitated.” Restoring to checkpoint values and repeating address the limitations of claim 26. Also see fig. 5B, #322, #333, and arrow to A (repeating command inputs).

In the remarks, the applicant has argued substantially that:

4) Regarding claim 28, the Kato reference does not teach the claimed steps of “creating a child process from the debug process”, “saving a process state of the child process” and “executing the child process using the stored process state”

Examiner’s response:

4) An example may be found at col. 8, lines 15-27, “...a function to manage a situation upon storage of a debugged state is further added. In this connection, storage situation management unit is additionally provided...upon production of state storage file (a child process), a file name and event condition which has made the cause of the storage of a debugged state into the file (a child process), a debugging object program name (a child process) ...are stored in a correlated condition into storage (a child process) ...” It is common for many child processes to be formed as a program executes. The claim language is not clear as to exactly what the child process does.

In the remarks, the applicant has argued substantially that:

Art Unit: 2122

5) Regarding claim 29, the Kato reference does not teach “saving a process state of a traced process and a process state of another process that is not being traced”.

Examiner's response:

Regarding claim 29, “saving a process state of the traced process and a process state of another process that is not being traced...” it is inherent that an executing computer program is saving information during execution (example: program counter), in addition to a program specific feature such as a debugging routine also saving debug / restore state.

Examiner maintains objections to claims 1-29.